AF

PATENT
Customer No. 58,982
Attorney Docket No. 08350.0243-00000

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of:	
Mohammed A. KHAN	Group Art Unit: 2123
Application No.: 10/024,359	Examiner: Ayal I. Sharon
Filed: December 21, 2001	
For: SYMBOLIC EVALUATION ENGINE FOR HIGH-PERFORMANCE SIMULATIONS	Confirmation No.: 6525

Mail Stop Appeal Brief--Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

APPEAL BRIEF UNDER BOARD RULE § 41.37

In support of the Notice of Appeal filed November 6, 2007, the period for response to which extends through January 7, 2008 (January 6, 2008 being a Sunday) and further to Board Rule 41.37, Appellant presents this brief. Please charge to our Deposit Account No. 06-0916 the Appeal Brief fee of $510.00 required under 37 C.F.R. § 1.17(c).

This Appeal responds to the final rejection of claims 1-60 in the Final Office Action mailed July 6, 2007.

If any additional fees are required or if the enclosed payment is insufficient, Appellant requests that the required fees be charged to Deposit Account No. 06-0916.

TABLE OF CONTENTS

REAL PARTY IN INTEREST

Caterpillar Inc. is the real party in interest.

RELATED APPEALS AND INTERFERENCES

There are currently no other appeals or interferences, of which Appellant, Appellant's legal representative, or Assignee are aware, that will directly affect or be directly affected by or have a bearing on the Board's decision in this pending appeal.

STATUS OF CLAIMS

Claims 1-60 are rejected under 35 U.S.C. § 101 as (i) allegedly preempting a 35 U.S.C. § 101 judicial exception and (ii) allegedly being drawn to non-statutory subject matter. The final rejections of claims 1-60 are being appealed. A copy of these claims is provided in the attached Claims Appendix to the Appeal Brief.

STATUS OF AMENDMENTS

No amendments have been filed subsequent to the claim amendments filed in the After Final Amendment on October 9, 2007, which were entered by the Examiner in the Advisory Action mailed on October 30, 2007.

SUMMARY OF CLAIMED SUBJECT MATTER

Independent claim 1 is drawn to a computer-implemented method of simulating a system. See, e.g., Appellant's Specification, as formatted in U.S. Patent Application Publication No. 2003/0120467, ("Specification") at ¶¶ 1, 7, and 101. The method includes establishing equations modeling the system using terms having characteristics encapsulated within the terms. See, e.g., Specification at ¶ 25. The method includes symbolically processing the established equations for simplification. See, e.g., Specification at ¶ 25. The symbolic processing includes utilizing the Pantelides algorithm to reduce the established equations and eliminating an integral, including assigning a preferred integration location rank to one or more integrals. See, e.g., Specification at ¶¶ 10, 11, 39, and 48-51; and Figs. 1, 2, and 4. The method includes processing system equations for efficient simulation, including processing a first set of equations including equations modeling the system and initial condition constraints and processing a second set of equations including equations modeling the system and numeric integration equations. See, e.g., Specification at ¶¶ 8, 9, 26, 27, 52, 53; and Fig. 1. The method includes simulating the system using the processed equations and displaying results of the simulation. See, e.g., Specification at ¶¶ 57, 87, 97, and 101.

Independent claim 23 is drawn to a machine-readable storage medium having stored thereon instructions executable by a computer to simulate a system. See, e.g., pecification at ¶¶ 1, 7, 101. The instructions include defining equations modeling the system using terms having characteristics encapsulated within the terms. See, e.g., Specification at ¶ 25. The instructions include symbolically processing the established equations for simplification. See, e.g., Specification at ¶ 25. The symbolic processing

includes utilizing the Pantelides algorithm to reduce the established equations and eliminating an integral, including assigning a preferred integration location rank to one or more integrals. See, e.g., Specification at ¶¶ 8, 9, 26, 27, 52, 53; and Figs. 1, 2, and 4. The instructions include processing system equations for efficient simulation, the system processing including processing a first set of equations including equations modeling the system and initial condition constraints and processing a second set of equations including equations modeling the system and numeric integration equations. See, e.g., Specification at ¶¶ 8, 9, 26, 27, 52, 53; and Fig. 1. The instructions also include simulating the system using the processed equations and displaying results of the simulation. See, e.g., Specification at ¶¶ 57, 87, 97, and 101.

Independent claim 43 is drawn to a computer-implemented method of simulating a system. See, e.g., Specification at ¶¶ 1, 7, 101. The method includes symbolically processing a set of equations that model the system, the symbolic processing including assigning a portion of the set of equations to variables that have non-zero partial derivatives and differentiating the remainder of the set of equations. See, e.g., Specification at ¶¶ 8, 9, and 39-41; and Fig. 2. The processing includes approximating an algebraic derivative for those equations that cannot be symbolically differentiated. See, e.g., Specification at ¶¶ 8, 9, 42; and Fig. 2. The processing includes symbolically integrating equations that cannot be assigned. See, e.g., Specification at ¶¶ 8, 9, 44; Fig. 2. The processing includes differentiating equations that add output derivates and integrating equations that add output integrals. See, e.g., Specification at ¶¶ 8, 9, 45; and Fig. 2. The processing further includes eliminating an integral as each symbolically differentiated or integrated equation eliminates a numeric integration, such that the

integral is converted to an algebraic variable by eliminating the derivative or integral relationship, wherein eliminating an integral includes assigning a preferred integration location rank to one or more integrals. See, e.g., Specification at ¶¶ 8, 9, 48; and Fig. 2. The method also includes simulating the system based on the symbolically processed set of equations and communicating results of the simulation to an external device. See, e.g., Specification at ¶¶ 57, 87, 97, and 101.

Independent claim 44 is drawn to a machine-readable storage medium having instructions stored thereon executable by a computer to simulate a system. See, e.g., Specification at ¶¶ 1, 7, 101. The instructions include symbolically processing a set of equations that model the system, the symbolic processing including assigning a portion of the set of equations to variables that have non-zero partial derivatives and differentiating the remainder of the set of equations. See, e.g., Specification at ¶¶ 8, 9, and 39-41; and Fig. 2. The processing includes approximating an algebraic derivative for those equations that cannot be symbolically differentiated. See, e.g., Specification at ¶¶ 8, 9, 42; and Fig. 2. The processing includes symbolically integrating equations that cannot be assigned. See, e.g., Specification at ¶¶ 8, 9, 44; and Fig. 2. The processing includes differentiating equations that add output derivates and integrating equations that add output integrals. See, e.g., Specification at ¶¶ 8, 9, 45; and Fig. 2. The processing further includes eliminating an integral as each symbolically differentiated or integrated equation eliminates a numeric integration, such that the integral is converted to an algebraic variable by eliminating the derivative or integral relationship, wherein eliminating an integral includes assigning a preferred integration location rank to one or more integrals. See, e.g., Specification at ¶¶ 8, 9, 48; and Fig.

2. The processing also includes simulating the system based on the symbolically processed set of equations and communicating results of the simulation to an external device. See, e.g., Specification at ¶¶ 57, 87, 97, and 101.

Independent claim 45 is drawn to a computer-implemented method of eliminating an integral in a Pantelides algorithm performed by a computer-based application that simulates a system. See, e.g., Specification at ¶¶ 1, 7, 39, 48, 97, and 101. The method includes assigning a preferred integration location rank to one or more integrals. See, e.g., Specification at ¶¶ 10, 11, 39, and 48-51; and Figs. 1, 2, and 4. The method includes utilizing the preferred integration location rank, assigning integrals to equations modeling the system. See, e.g., Specification at ¶¶ 48-51; and Figs. 1, 2, and 4. The method includes eliminating from the equations the integration of assigned or solved integral variables. See, e.g., Specification at ¶¶ 51; and Figs. 1, 2, and 4. The method also includes simulating the system using the equations and displaying results of the simulation. See, e.g., Specification at ¶¶ 57, 87, 97, and 101.

Independent claim 49 is drawn to a machine-readable storage medium having stored thereon instructions executable by a computer to eliminate an integral in a Pantelides algorithm in an application that simulates a system. See, e.g., Specification at ¶¶ 1, 7, 39, 48, 97, and 10. The instructions include assigning a preferred integration location rank to one or more integrals. See, e.g., Specification at ¶¶ 10, 11, 39, and 48-51; and Figs. 1, 2, and 4. The instructions include utilizing the preferred integration location rank, assigning integrals to equations defining the system. See, e.g., Specification at ¶¶ 48-51; and Figs. 1, 2, and 4. The instructions include eliminating from the equations the integration of assigned or solved integral variables. See, e.g.,

Specification at ¶¶ 51; and Figs. 1, 2, and 4. The instructions also include simulating the system using the equations and displaying results of the simulation. See, e.g., Specification at ¶¶ 57, 87, 97, and 101.

Independent claim 53 is drawn to a computer-implemented method of simulating systems. See, e.g., Specification at ¶¶ 1, 7, 101. The method including performing a tearing process on a set of equations modeling a system. See, e.g., Specification at ¶¶ 12, 13, and 70; and Fig. 6. The method includes identifying block variables in the equations in a block in which the block variables appear linearly with constant coefficients. See, e.g., Specification at ¶ 72; and Fig. 6. The method including determining the solvability of the nonlinear equations. See, e.g., Specification at ¶ 75; and Fig. 6. The method includes solving nonlinear integration equations for their respective integrals. See, e.g., Specification at ¶¶ 73 and 86; and Fig. 6. The method includes solving the linear equations. See, e.g., Specification at ¶ 74; and Fig. 6. The method includes solving the nonlinear equations utilizing iterates and block variables solved from the linear equations. See, e.g., Specification at ¶ 76; and Fig. 6. The method includes scanning for solved for variables identification of variables that are independent and may be removed from the block. See, e.g., Specification at ¶ 81; and Fig. 6. The method also includes simulating the system using the processed equations and displaying results of the simulation. See, e.g., Specification at ¶¶ 57, 87, 97, and 101.

Independent claim 54 is drawn to a machine-readable storage medium having stored thereon instructions executable by a computer to simulate a system. See, e.g., Specification at ¶¶ 1, 7, 39, 48, 97, and 10. The instructions include performing a

tearing process on a set of equations modeling a system. See, e.g., Specification at ¶¶ 12, 13, and 70; and Fig. 6. The instructions include identifying block variables in the equations in a block in which the block variables appear linearly with constant coefficients. See, e.g., Specification at ¶ 72; and Fig. 6. The instructions include determining the solvability of the nonlinear equations. See, e.g., Specification at ¶ 75; and Fig. 6. The instructions include solving nonlinear integration equations for their respective integrals. See, e.g., Specification at ¶¶ 73 and 86; and Fig. 6. The instructions include solving the linear equations. See, e.g., Specification at ¶ 74; and Fig. 6. The instructions include solving the nonlinear equations utilizing iterates and block variables solved from the linear equations. See, e.g., Specification at ¶ 76; and Fig. 6. The instructions include scanning for solved variables for identification of variables that are independent and may be removed from the block. See, e.g., Specification at ¶ 81; and Fig. 6. The instructions also include simulating the system using the processed equations and displaying results of the simulation. See, e.g., Specification at ¶¶ 57, 87, 97, and 101.

Independent claim 58 is drawn to a computer-implemented method of simulating a component of a system. See, e.g., Specification at ¶¶ 1, 7, 15, 28, 33, 34, and 38. The method includes establishing equations modeling the component using terms having characteristics encapsulated within the terms. See, e.g., Specification at ¶ 26; and Fig. 1. The method includes symbolically processing the established equations for simplification. See, e.g., Specification at ¶ 25. The symbolic processing reduces the established equations by eliminating an integral, wherein eliminating an integral includes assigning a preferred integration location rank to one or more integrals. See,

e.g., Specification at ¶¶ 10, 11, 39, and 48-51; and Figs. 1, 2, and 4. The method includes processing system equations for efficient simulation, the equation processing including processing a first set of equations including equations modeling the component and initial condition constraints and processing a second set of equations including equations modeling the component and numeric integration equations. See, e.g., Specification at ¶¶ 8, 9, 26, 27, 52, 53; and Fig. 1. The method also includes simulating the component using the processed equations and displaying results of the simulation. See, e.g., Specification at ¶¶ 57, 87, 97, and 101.

Independent claim 59 is drawn to a computer-implemented method of simulating a system. See, e.g., Specification at ¶¶ 1, 7, 101. The method includes establishing equations modeling the system. See, e.g., Specification at ¶ 25. The method includes symbolically processing the established equations for simplification, wherein the symbolic processing reduces the established equations by eliminating an integral and eliminating an integral includes assigning a preferred integration location rank to one or more integrals. See, e.g., Specification at ¶¶ 10, 11, 39, and 48-51; and Figs. 1, 2, and 4. The method includes establishing a first set of equations including equations modeling the system and initial condition constraints and establishing a second set of equations including equations modeling the system and numeric integration equations that constrain integrated variables. See, e.g., Specification at ¶¶ 8, 9, 26, 27, 52, 53; and Fig. 1. The method includes processing the first and second sets of equations independently and in parallel, to generate initial condition and transient solutions. See, e.g., Specification at ¶¶ 27; and Fig. 1. The method also includes simulating the

system using the processed equations and displaying results of the simulation. See, .g., Specification at ¶¶ 57, 87, 97, and 101.

GROUNDS OF REJECTION

A. Claims 1-60 stand rejected under 35 U.S.C. § 101 as allegedly preempting every substantial practical application of a 35 U.S.C. § 101 judicial exception.

B. Claims 1-60 stand rejected under 35 U.S.C. § 101 as allegedly being drawn to non-statutory subject matter.

ARGUMENT

A. The rejection of claims 1-60 under 35 U.S.C. § 101 as allegedly preempting every substantial practical application of a 35 U.S.C. § 101 judicial exception should be reversed.

Appellant respectfully requests that the Board reverse the Examiner's rejection of claims 1-60 under 35 U.S.C. § 101 as preempting every substantial practical application of a 35 U.S.C. § 101 judicial exception. Final Office Action at 2. Appellant respectfully submits that claims 1-60 would not preempt every substantial practical application of a 35 U.S.C. § 101 judicial exception.

1. Claims 1-42, 45-54, and 56-60

Claims 1-42, 45-54, and 56-60 are drawn to computer-implemented methods, or machine-readable storage devices containing computer-executable instructions, for establishing and processing equations that model a system in a specific manner, using the processed equations to simulate the system, and displaying the results of the simulation (i.e., a computer simulation tool). Thus, Appellant's claims cannot preempt *every* substantial practical application of a judicial exception, because *simulation does not constitute every practical application of a 35 U.S.C. § 101 judicial exception under 35 U.S.C. § 101, but rather, one practical application thereof.*

The Examiner cites Gottschalk v. Benson, 409 U.S. (1972), in support of the rejection. Final Office Action at 2. The issue decided in Benson, however, does not support the Examiner's contention that Appellant's claims preempt every substantial practical application of 35 U.S.C. § 101 judicial exception. As noted by the Examiner at page 2 of the Final Office Action, in Benson the Supreme Court held that a claim reciting a computer that solely calculates a mathematical formula or a computer disk

that solely stores a mathematical formula is an unpatentable attempt to patent the mathematical formula itself. Gottschalk v. Benson, 409 U.S. 63, at 71-72. However, this Board's reviewing court has held many times that mathematical operations performed by computers do not automatically negate the patentability of processes that use those operations. Arrhythmia Research Tech. Inc. v. Corazonix Corp., 958 F.2d 1053 (Fed. Cir. 1992). Further, even if a claim would read on a general purpose computer programmed to carry out the claimed invention, the Federal Circuit has held that "such programming creates a new machine, because a general purpose computer in effect becomes a special purpose computer once it is programmed to perform particular functions pursuant to instructions from program software." In re Alappat, 33 F.3d 1526, 1545 (Fed. Cir. 1994).

Thus, the Appellant's claims drawn to a computer-implemented method of simulating a system including processing equations modeling the system for simplification would not preempt every application of the mathematical functions recited. For example, *the claims further recite that the system is simulated using the processed equations and that results of the simulation are displayed*. Like the claims in Arrhythmia Research, the present claims "do not seek to patent a mathematical formula . . . they seek only to foreclose from others the use of that equation in conjunction with all of the other steps in [his] claimed process." Arrhythmia Research, 958 F.2d at 1054 citing Diamond v. Diehr, 450 U.S. 175 (1981).

2. Claims 43, 44, and 55-57

Similarly, claims 43, 44, and 55-57 are drawn to computer-implemented methods, or machine-readable storage devices containing computer-executable

instructions, for establishing and processing equations that model a system in a specific manner, using the processed equations to simulate the system, and communicating results of the simulation to another device (i.e., a computer simulation tool). Thus, claims 43, 44, and 55-57 also cannot preempt *every* substantial practical application of a 35 U.S.C. § 101 judicial exception, because they are drawn to establishing and processing equations in conjunction with all the other claim elements. Appellant submits that simulation does not constitute *every* practical application of a 35 U.S.C. § 101 judicial exception, but rather, *one* practical application thereof.

For at least the foregoing reasons, Appellant submits that Appellant's claims 1-60 do not preempt every substantial practical application of a 35 U.S.C. § 101 judicial exception. Appellant requests the reversal of the rejection of claims 1-60 under 35 U.S.C. § 101 as allegedly preempting every substantial practical application of a 35 U.S.C. § 101 judicial exception.

B. The rejection of claims 1-60 under 35 U.S.C. § 101 as allegedly being drawn to non-statutory subject matter should be reversed.

Appellant requests the reversal of the rejection of claims 1-60 under 35 U.S.C. § 101 as being directed to nonstatutory subject matter for lacking a "concrete, useful, and tangible" result. Final Office Action at 3. Claims 1-60 recite a "concrete, useful, and tangible" result.

1. Claims 1-42, 45-54, and 56-60 recite "useful, concrete, and tangible" results.

One test for determining whether a process is statutory is to determine whether it produces a "useful, concrete, and tangible result." State Street Bank & Trust Co. v. Signature Financial Group, Inc., 149 F.3d 1368, 1373 (Fed. Cir. 1998.). Recently, in Ex

Parte Bernard et al., 2007 WL 2183647 (Bd.Pat.App. & Interf. 2007), this Board held statutory claims directed to the transformation of data via extraction and processing. Id. at 3. Specifically, the Board reasoned that the "transformation of intangible subject matter (i.e., data or signals) by a computer may qualify as a § 101 process," and that "displaying information" is a useful, concrete, and tangible result. Id. at 3-4.

Like the claims held to be statutory in Ex Parte Bernard et al., claims 1-42, 45-54, and 56-60 are directed to the computer-based transformation and display of data. Specifically, these claims are drawn to computer-implemented methods, or computer-readable storage media containing computer-executable instructions, for processing equations that model a system, simulating the system using the processed equations, and displaying results of the simulation. Modeling a system with equations, processing the equations in a particular manner, and using the processed equations to simulate the system constitutes a "transformation of intangible subject matter (i.e., data or signals)," as contemplated by the Board in Ex Parte Bernard et al. Specifically, the equations modeling the system are transformed from a first form, in which simulation of the system is unduly burdensome due to the large amount of computational resources required for simulation, and a second, processed form, in which simulation can be done more efficiently. Specification at ¶¶ 0005, 0082, 0086, 0094, and 0102. The claimed processed equations themselves constitute a "useful, concrete, and tangible" result because they reduce the computational effort and resources required to simulate the system and enhance simulation performance. Specification at ¶ 0005 and 0102.

In addition, displaying results of the simulation is a useful, concrete, and tangible result in view of Ex Parte Bernard et al. For example, a computer tool that displays the

results of a simulation allows an engineer to determine if a particular system design will perform up to expectations or if changes to the design are required. In addition, the U.S. Patent Office acknowledges that simulation, as an application, is useful. This is evidenced by the existence of class 703, which awards patents for inventions drawn to simulation, modeling, and emulation applications.

2. Claims 43, 44, and 55-57 also recite "useful, concrete, and tangible" results.

Likewise, claims 43, 44, and 55-57 are directed to computer-implemented methods, or machine-readable storage devices containing computer-executable instructions, for processing equations that model a system, simulating the system using the processed equations, and communicating results of the simulation to an external device. Processing equations that model a system and using the processed equations to simulate the system constitute a "transformation of intangible subject matter (i.e., data or signals)," as contemplated by this Board in Ex Parte Bernard et al. Specifically, the equations modeling the system are transformed from a first form, in which simulation of the system is unduly burdensome due to the large amount of computational resources required for simulation, and a second, processed form, in which simulation can be done more efficiently. Specification at ¶¶ 0005, 0082, 0086, 0094, and 0102. The claimed processed equations themselves constitute a "useful, concrete, and tangible" result because they reduce the computational effort and resources required to simulate the system and enhance simulation performance. Specification at ¶ 0005 and 0102.

Further, communicating the simulation results to another device is a useful, concrete, and tangible result in view of Ex Parte Bernard et al. For example, the communication of simulation results to an external device allows the simulation results to be stored and viewed by others. In addition, the U.S. Patent Office acknowledges that simulation, as an application, is useful. This is evidenced by the existence of class 703, which awards patents for inventions drawn to simulation, modeling, and emulation applications.

3. Claims 1-60 recite statutory subject matter even under the most recent federal case law because they are "tied to" machine.

In addition to reciting statutory processes under § 101 directed to a practical application (i.e., simulation) of a process that produces "useful, concrete, and tangible" results Appellant's claims are also statutory under § 101 because they are "embodied in, operate[] on, transform[], or otherwise involve[] a . . . machine." In re Comiskey, 499 F.3d 1365, 1374 (Fed. Cir. 2007). Specifically, the Federal Circuit noted that "the Supreme Court has held that a claim reciting an algorithm or abstract idea can be statutory subject matter only if, as employed in the process, it is embodied in, operates, transforms, or otherwise involves another class of statutory subject matter, i.e., a machine . . ." Id.

Independent claims 1, 43, 45, 53, 55, 58, and 59 recite *computer-implemented* methods of simulating a system and displaying or communicating results of the simulation. As such, these claims, and their respective dependent claims, are "tied to," or "operate on," a computer, making them statutory under In re Comiskey.

Similarly, the remaining independent claims 23, 44, 49, and 54 and their respective dependent claims recite *machine-readable storage media* storing *computer-executable* instructions for simulating a system. Even if these claims recite abstract ideas (which Appellant does not admit), these claims also recite methods or algorithms that are "tied to" or "embodied in" statutory subject matter (i.e., machine-readable storage media), and are therefore statutory under In re Comiskey.

For at least the foregoing reasons, Appellant submits that claims 1-60 are drawn to statutory subject matter under 35 U.S.C. § 101. Appellant therefore requests the reversal of the rejection of claims 1-60 under 35 U.S.C. § 101 as being drawn to non-statutory subject matter.

E. Conclusion

For the reasons given above, Appellant's claims 1-60: (1) produce "useful, concrete, and tangible" results; (2) do not preempt every substantial practical application of a 35 U.S.C. § 101 judicial exception; and (3) are "tied to" or "embodied in" another class of statutory subject matter. Claims 1-60 are thus drawn to patentable subject matter under 35 U.S.C. § 101. Appellant respectfully requests the reversal of the rejections of claims 1-60 under 35 U.S.C. § 101.

To the extent any extension of time under 37 C.F.R. § 1.136 is required to obtain entry of this Appeal Brief, such extension is hereby respectfully requested. If there are any fees due under 37 C.F.R. §§ 1.16 or 1.17 which are not enclosed herewith, including any fees required for an extension of time under 37 C.F.R. § 1.136, please charge such fees to Deposit Account No. 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: January 7, 2007

By: ____________________
Erika H. Arner
Reg. No. 57,540

Claims Appendix to Appeal Brief Under Rule 41.37(c)(1)(viii)

1. (Previously presented) A computer-implemented method of simulating a system, comprising:

establishing equations modeling the system using terms having characteristics encapsulated within the terms;

symbolically processing the established equations for simplification, wherein the symbolic processing includes:

utilizing the Pantelides algorithm to reduce the established equations; and

eliminating an integral, wherein eliminating an integral includes assigning a preferred integration location rank to one or more integrals;

processing system equations for efficient simulation, wherein processing the system equations includes:

processing a first set of equations including equations modeling the system and initial condition constraints; and

processing a second set of equations including equations modeling the system and numeric integration equations;

simulating the system using the processed equations; and

displaying results of the simulation.

2. (Original) The method of claim 1, wherein the stage of defining equations further includes:

defining equations modeling the system using terms selected from one or more basic terms, composite terms, or collection terms.

3. (Original) The method of claim 1, further including:

extending a library of terms by defining new term classes, wherein term classes define objects having characteristics encapsulated within the objects.

4. (Original) The method of claim 1, further including:

defining a term group including one or more terms having related functionality;

evaluating each term within the term group upon an initial request for evaluation of any of the one or more terms within the term group;

storing the result of the evaluation for each of the one or more terms within the term group; and

recalling the stored value of the evaluated one or more terms from the term group upon a subsequent request for evaluation of the one or more terms, without performing the evaluation stage.

5. (Previously presented) The method of claim 1, wherein utilizing the Pantelides algorithm includes reducing the established equations to a system of equations having a differential-algebraic system of equations index of at most one.

6. (Original) The method of claim 5, wherein utilizing the Pantelides algorithm further includes:

assigning equations to variables that have non-zero partial derivatives; and

differentiating the remainder of the equations.

7. (Previously presented) The method of claim 5, wherein utilizing the Pantelides algorithm further includes:

approximating an algebraic derivative for those equations that cannot be symbolically differentiated.

8. (Original) The method of claim 5, wherein utilizing the Pantelides algorithm further includes:

symbolically integrating equations that cannot be assigned.

9. (Original) The method of claim 5, wherein utilizing the Pantelides algorithm further includes:

differentiating equations that add output derivates and integrating equations that add output integrals.

10. (Previously presented) The method of claim 5, wherein eliminating an integral further includes:

eliminating an integral as each symbolically differentiated or integrated equation eliminates a numeric integration, such that the integral is converted to an algebraic variable by eliminating the derivative or integral relationship.

11. (Previously presented) The method of claim 10, wherein eliminating an integral further includes:

utilizing the preferred integration location rank, assigning integrals to equations; and

eliminating the integration of assigned or solved integral variables.

12. (Previously presented) The method of claim 1, wherein assigning a preferred integration location rank further includes:

assigning a preferred integration location to one or more integrals, a user assigned preferred integration location being given the highest available preferred integration location rank;

assigning a preferred integration location rank, wherein the preferred integration location rank has a lower rank than the user defined preferred integration location rank; and

assigning all other integration locations a default lowest rank.

13. (Original) The method of claim 12, wherein the assigned preferred integration location is assigned by a user.

14. (Original) The method of claim 12, wherein the assigned preferred integration location rank is assigned by a component developer.

15. (Original) The method of claim 12, wherein utilizing the preferred integration location ranks to assign integrals to equations further includes:

identifying integral variables that appear linearly and nonlinearly in the integral equations;

establishing a current preferred integration location rank at a default setting;

assigning each integral equation an integral variable that has a preferred integration location rank of less than the current preferred

integration location rank, and, if possible, appears linearly in the equation; and

repeating the previous three stages after increasing the current preferred integration location rank until a maximum preferred integration location rank has been exceeded.

16. (Previously presented) The method of claim 15, further including:

solving each integral equation that is assigned an integral variable that appears linearly in the integral equation;

substituting the solved value into other equations; and

if due to substitutions, an one of the assigned variables is no longer in the equation, assign another integral with minimum integration rank to the one of the assigned variables.

17. (Previously Presented) The method of claim 1, wherein the stage of performing system processing includes:

establishing an initial condition system using the first set of equations and establishing a transient system using the second set of equations.

18. (Previously presented) The method of claim 1, wherein processing a first set of equations includes:

processing a first set of equations including equations modeling the system and user-defined and component-defined initial condition constraints.

19. (Previously presented) The method of claim 1, wherein performing system processing includes:

performing the system processing on the first set of equations and the second set of equations independently and in parallel.

20. (Original) The method of claim 1, wherein system processing further includes:

replacing alias variables;

partitioning the equations into blocks;

tearing the blocks;

sorting the blocks; and

compressing equation terms.

21. (Original) The method of claim 20, wherein tearing the equations includes:

identifying block variables in the equations in the block in which the block variables appear linearly with constant coefficients;

solving nonlinear integration equations for their respective integrals;

solving the linear equations;

determining the solvability of the nonlinear equations;

solving the nonlinear equations utilizing iterates and block variables solved from the linear equations; and

scanning the solved variables for identification of variables that are independent and may be removed from the block.

22. (Original) The method of claim 20, wherein block sorting further includes:

defining and identifying the blocks as static blocks, dynamic blocks, or output blocks;

removing the static blocks from a list of blocks; and

removing the output blocks from the list of blocks.

23. (Previously presented) A machine-readable storage medium having stored thereon instructions executable by a computer to simulate a system, the instructions comprising:

defining equations modeling the system using terms having characteristics encapsulated within the terms;

symbolically processing the established equations for simplification, wherein the symbolic processing includes:

utilizing the Pantelides algorithm to reduce the established equations; and

eliminating an integral, wherein eliminating an integral includes assigning a preferred integration location rank to one or more integrals;

processing system equations for efficient simulation, wherein processing the system equations includes:

processing a first set of equations including equations modeling the system and initial condition constraints; and

processing a second set of equations including equations modeling the system and numeric integration equations;

simulating the system using the processed equations; and

displaying results of the simulation.

24. (Previously presented) The machine-readable storage medium of claim 23, wherein defining equations further includes:

defining equations modeling the system using terms selected from one or more basic terms, composite terms, or collection terms.

25. (Previously presented) The machine-readable storage medium of claim 23, wherein the instructions further include:

extending a library of terms by defining new term classes, wherein term classes define objects having characteristics encapsulated within the objects.

26. (Previously presented) The machine-readable storage medium of claim 23, wherein the instructions further include:

defining a term group including one or more terms having related functionality;

evaluating each term within the term group upon an initial request for evaluation of any of the one or more terms within the term group;

storing the result of the evaluation for each of the one or more terms within the term group; and

recalling the stored value of the evaluated one or more terms from the term group upon a subsequent request for evaluation of the one or more terms, without performing the evaluation stage.

27. (Previously presented) The machine-readable storage medium of claim 23, wherein utilizing the Pantelides algorithm includes reducing the established equations to a system of equations having a differential-algebraic system of equations index of at most one.

28. (Original) The machine-readable storage medium of claim 27, wherein utilizing the Pantelides algorithm further includes:

assigning equations to variables that have non-zero partial derivatives; and

differentiating the remainder of the equations.

29. (Previously presented) The machine-readable storage medium of claim 27, wherein utilizing the Pantelides algorithm further includes:

approximating an algebraic derivative for those equations that cannot be symbolically differentiated.

30. (Original) The machine-readable storage medium of claim 27, wherein utilizing the Pantelides algorithm further includes:

symbolically integrating equations that cannot be assigned.

31. (Original) The machine-readable storage medium of claim 27, wherein utilizing the Pantelides algorithm further includes:

differentiating equations that add output derivates and integrating equations that add output integrals.

32. (Previously presented) The machine-readable storage medium of claim 27, wherein eliminating an integral further includes:

eliminating an integral as each symbolically differentiated or integrated equation eliminates a numeric integration, such that the integral is converted to an algebraic variable by eliminating the derivative or integral relationship.

33. (Previously presented) The machine-readable storage medium of claim 1, wherein eliminating an integral further includes:

utilizing the preferred integration location rank, assigning integrals to equations; and

eliminating the integration of assigned or solved integral variables.

34. (Previously presented) The machine-readable storage medium of claim 23, wherein assigning a preferred integration location rank further includes:

assigning, by a user, a preferred integration location to one or more integrals, the user assigned preferred integration location being given the highest available preferred integration location rank;

assigning, by a component developer, a preferred integration location rank, wherein the preferred integration location rank has a lower rank than the user defined preferred integration location rank; and

assigning all other integration locations a default lowest rank.

35. (Original) The machine-readable storage medium of claim 34, wherein utilizing the preferred integration location ranks to assign integrals to equations, further includes:

identifying integral variables that appear linearly and nonlinearly in the integral equations;

establishing a current preferred integration location rank at a default setting;

assigning each integral equation an integral variable that has a preferred integration location rank of less than the current preferred integration location rank and, if possible, appears linearly in the equation; and

repeating the previous three stages after increasing the current preferred integration location rank until a maximum preferred integration location rank has been exceeded.

36. (Previously presented) The machine-readable storage medium of claim 35, wherein the instructions further include:

solving each integral equation that is assigned an integral variable that appears linearly in the integral equation;

substituting the solved value into other equations; and

if due to substitutions, an one of the assigned variables is no longer in the equation, assign another integral with minimum integration rank to the one of the assigned variables.

37. (Previously presented) The machine-readable storage medium of claim 23, wherein the stage of performing system processing includes:

establishing an initial condition system using the first set of equations and establishing a transient system using the second set of equations.

38. (Previously presented) The machine-readable storage medium of claim 23, wherein processing a first set of equations includes:

processing a first set of equations including equations modeling the system and user-defined and component-defined initial condition constraints.

39. (Previously presented) The machine-readable storage medium of claim 23, wherein performing system processing includes:

performing the system processing on the first set of equations and the second set of equations independently and in parallel.

40. (Previously presented) The machine-readable storage medium of claim 23, wherein processing system equations further includes:

replacing alias variables;

partitioning the equations into blocks;

tearing the blocks;

sorting the blocks; and

compressing equation terms.

41. (Original) The machine-readable storage medium of claim 40, wherein tearing the block includes:

identifying block variables in the equations in the block in which the block variables appear linearly with constant coefficients;

solving nonlinear integration equations for their respective integrals;

determining the solvability of the nonlinear equations;

solving the nonlinear equations utilizing iterates and block variables solved from the linear equations;

solving the linear equations; and

scanning the solved variables for identification of variables that are independent and may be removed from the block.

42. (Original) The machine-readable storage medium of claim 40, wherein block sorting further includes:

defining and identifying the blocks as static blocks, dynamic blocks, or output blocks;

removing the static blocks from a list of blocks; and

removing the output blocks from the list of blocks.

43. (Previously presented) A computer-implemented method of simulating a system, comprising:

symbolically processing a set of equations that model the system, including:

assigning a portion of the set of equations to variables that have non-zero partial derivatives;

differentiating the remainder of the set of equations;

approximating an algebraic derivative for those equations that cannot be symbolically differentiated;

symbolically integrating equations that cannot be assigned;

differentiating equations that add output derivates and integrating equations that add output integrals;

eliminating an integral as each symbolically differentiated or integrated equation eliminates a numeric integration, such that the integral is converted to an algebraic variable by eliminating the derivative or integral relationship, wherein eliminating an integral includes assigning a preferred integration location rank to one or more integrals;

simulating the system based on the symbolically processed set of equations; and

communicating results of the simulation to an external device.

44. (Previously presented) A machine-readable storage medium having stored thereon executable by a computer to simulate a system, the instructions comprising:

symbolically processing a set of equations, including:

assigning a portion of the set of equations to variables that have non-zero partial derivatives;

differentiating the remainder of the set of equations;

approximating an algebraic derivative for those equations that cannot be symbolically differentiated;

symbolically integrating equations that cannot be assigned;

differentiating equations that add output derivates and integrating equations that add output integrals;

eliminating an integral as each symbolically differentiated or integrated equation eliminates a numeric integration, such that the integral is converted to an algebraic variable by eliminating the derivative or integral relationship, wherein eliminating an integral includes assigning a preferred integration location rank to one or more integrals;

simulating the system based on the symbolically processed set of equations; and

communicating results of the simulation to an external device.

45. (Previously presented) A computer-implemented method of eliminating an integral in a Pantelides algorithm performed by a computer-based application that simulates a system, comprising:

assigning a preferred integration location rank to one or more integrals;

utilizing the preferred integration location rank, assigning integrals to equations modeling the system;

eliminating from the equations the integration of assigned or solved integral variables;

simulating the system using the equations; and

displaying results of the simulation.

46. (Original) The method of claim 45, wherein assigning a preferred integration location rank, further includes:

assigning, by a user, a preferred integration location to one or more integrals, the user assigned preferred integration location being given the highest available preferred integration location rank;

assigning, by a component developer, a preferred integration location rank, wherein the preferred integration location rank has a lower rank than the user defined preferred integration location rank; and

assigning all other integration locations a default lowest rank.

47. (Original) The method of claim 46, wherein utilizing the preferred integration location ranks to assign integrals to equations, further includes:

identifying integral variables that appear linearly and nonlinearly in the integral equations;

establishing a current preferred integration location rank at a default setting;

assigning each integral equation an integral variable that has a preferred integration location rank of less than the current preferred integration location rank and, if possible, appears linearly in the equation; and

repeating the previous three stages after increasing the current preferred integration location rank until a maximum preferred integration location rank has been exceeded.

48. (Previously presented) The method of claim 47, further including:

solving each integral equation that is assigned an integral variable that appears linearly in the integral equation;

substituting the solved value into other equations; and

if due to substitutions, an one of the assigned variables is no longer in the equation, assign another integral with minimum integration rank to the one of the assigned variables.

49. (Previously presented) A machine-readable storage medium having stored thereon instructions executable by a computer to eliminate an integral in a Pantelides algorithm in an application that simulates a system, the instructions comprising:

assigning a preferred integration location rank to one or more integrals;

utilizing the preferred integration location rank, assigning integrals to equations defining the system;

eliminating from the equations the integration of assigned or solved integral variables;

simulating the system using the equations; and

displaying results of the simulation.

50. (Original) The machine-readable storage medium of claim 49, wherein assigning a preferred integration location rank, further includes:

assigning, by a user, a preferred integration location to one or more integrals, the user assigned preferred integration location being given the highest available preferred integration location rank;

assigning, by a component developer, a preferred integration location rank, wherein the preferred integration location rank has a lower rank than the user defined preferred integration location rank; and

assigning all other integration locations a default lowest rank.

51. (Previously presented) The machine-readable storage medium of claim 50, wherein utilizing the preferred integration location ranks to assign integrals to equations further includes:

identifying integral variables that appear linearly and nonlinearly in the integral equations;

establishing a current preferred integration location rank at a default setting;

assigning each integral equation an integral variable that has a preferred integration location rank of less than the current preferred integration location rank and, if possible, appears linearly in the equation; and

repeating the previous three stages after increasing the current preferred integration location rank until a maximum preferred integration location rank has been exceeded.

52. (Previously presented) The machine-readable storage medium of claim 51, wherein the instructions further include:

solving each integral equation that is assigned an integral variable that appears linearly in the integral equation;

substituting the solved value into other equations; and

if due to substitutions, an one of the assigned variables is no longer in the equation, assign another integral with minimum integration rank to the one of the assigned variables.

53. (Previously presented) A computer-implemented method of simulating systems, comprising:

performing a tearing process on a set of equations modeling a system, including:

identifying block variables in the equations in a block in which the block variables appear linearly with constant coefficients;

determining the solvability of the nonlinear equations;

solving nonlinear integration equations for their respective integrals;

solving the linear equations;

solving the nonlinear equations utilizing iterates and block variables solved from the linear equations;

scanning for solved for variables for identification of variables that are independent and may be removed from the block;

simulating the system using the processed equations; and

displaying results of the simulation.

54. (Previously presented) A machine-readable storage medium having stored thereon instructions executable by a computer to simulate a system, the instructions comprising:

performing a tearing process on a set of equations modeling a system, including:

identifying block variables in the equations in a block in which the block variables appear linearly with constant coefficients;

solving nonlinear integration equations for their respective integrals;

solving the linear equations;

determining the solvability of the nonlinear equations;

solving the nonlinear equations utilizing iterates and block variables solved from the linear equations;

scanning for solved for variables for identification of variables that are independent and may be removed from the block;

simulating the system using the processed equations; and

displaying results of the simulation.

55. (Previously presented) A computer-implemented method of simulating a system, comprising:

establishing equations modeling the system using terms having characteristics encapsulated within the terms;

symbolically processing the established equations for reducing the number of terms in the equations, wherein the symbolic processing reduces the established equations by eliminating an integral, wherein eliminating an integral includes assigning a preferred integration location rank to one or more integrals;

processing system equations for efficient simulation, wherein processing the system equations includes:

processing a first set of equations including equations modeling the system and initial condition constraints; and

processing a second set of equations including equations modeling the system and numeric integration equations;

simulating the system using the processed equations; and

communicating results of the simulation to an external device.

56. (Original) The method of claim 55, further including:

defining a term group including one or more terms having related functionality;

evaluating each term within the term group upon an initial request for evaluation of any of the one or more terms within the term group; and

storing the result of the evaluation fro each of the one or more terms within the term group.

57. (Original) The method of claim 56, further including:

recalling the stored value of the evaluated one or more terms from the term group upon a subsequent request for evaluation of the one or more terms, without performing the evaluation stage.

58. (Previously presented) A computer-implemented method of simulating a component of a system, comprising:

establishing equations modeling the component using terms having characteristics encapsulated within the terms;

symbolically processing the established equations for simplification, wherein the symbolic processing reduces the established equations by eliminating an integral, wherein eliminating an integral includes assigning a preferred integration location rank to one or more integrals;

processing system equations for efficient simulation, wherein processing system equations includes:

processing a first set of equations including equations modeling the component and initial condition constraints; and

processing a second set of equations including equations modeling the component and numeric integration equations;

simulating the component using the processed equations; and

displaying results of the simulation.

59. (Previously presented) A computer-implemented method of simulating a system, comprising:

establishing equations modeling the system;

symbolically processing the established equations for simplification, wherein the symbolic processing reduces the established equations by eliminating an integral, wherein eliminating an integral includes assigning a preferred integration location rank to one or more integrals;

establishing a first set of equations including equations modeling the system and initial condition constraints;

establishing a second set of equations including equations modeling the system and numeric integration equations that constrain integrated variables;

processing the first and second sets of equations independently and in parallel, to generate initial condition and transient solutions; and

simulating the system using the processed equations; and

displaying results of the simulation.

60. (Previously presented) The method of claim 59, wherein establishing equations modeling the system comprises establishing component equations, connectivity equations, and boundary condition equations; and wherein each of the first and second set of equations includes component, connectivity, and boundary condition equations.

Evidence Appendix to Appeal Brief Under Rule 41.37(c)(1)(ix)

There is no evidence being cited by Appellant in this Appeal.

Related Proceedings Appendix to Appeal Brief Under Rule 41.37(c)(1)(x)

To Appellant's knowledge, there are no related proceedings.